VMSLIB

```
SSSSSSSS     CCCCCCCC  RRRRRRRR   VV     VV  EEEEEEEEEE   CCCCCCCC  TTTTTTTTTT   000000     RRRRRRRR
SSSSSSSS     CCCCCCCC  RRRRRRRR   VV     VV  EEEEEEEEEE   CCCCCCCC  TTTTTTTTTT   000000     RRRRRRRR
SS           CC        RR     RR  VV     VV  EE           CC           TT       00    00    RR     RR
SS           CC        RR     RR  VV     VV  EE           CC           TT       00    00    RR     RR
SS           CC        RR     RR  VV     VV  EE           CC           TT       00    00    RR     RR
SS           CC        RR     RR  VV     VV  EE           CC           TT       00    00    RR     RR
SSSSSS       CC        RRRRRRRR   VV     VV  EEEEEEE      CC           TT       00    00    RRRRRRRR
SSSSSS       CC        RRRRRRR    VV     VV  EEEEEEE      CC           TT       00    00    RRRRRRR
      SS     CC        RR   RR    VV     VV  EE           CC           TT       00    00    RR   RR
      SS     CC        RR    RR   VV     VV  EE           CC           TT       00    00    RR    RR
      SS     CC        RR     RR   VV   VV   EE           CC           TT       00    00    RR     RR     ....
      SS     CC        RR     RR   VV   VV   EE           CC           TT       00    00    RR     RR     ....
SSSSSSSS     CCCCCCCC  RR     RR    VV VV    EEEEEEEEEE   CCCCCCCC     TT        000000     RR     RR     ....
SSSSSSSS     CCCCCCCC  RR     RR     VV      EEEEEEEEEE   CCCCCCCC     TT        000000     RR     RR     ....


LL           IIIIII     SSSSSSSS
LL           IIIIII     SSSSSSSS
LL             II     SS
LL             II     SS
LL             II     SS
LL             II     SS
LL             II       SSSSSS
LL             II       SSSSSS
LL             II             SS
LL             II             SS
LL             II             SS
LL             II             SS
LLLLLLLLLL   IIIIII     SSSSSSSS
LLLLLLLLLL   IIIIII     SSSSSSSS
```

B 13

SCR$VECTOR   - Entry vectors for Screen Package   16-SEP-1984 02:16:59   VAX/VMS Macro V04-00   Page 1
V04-000   5-SEP-1984 04:43:38   [VMSLIB.SRC]SCRVECTOR.MAR;1   (1)

```
0000      1                    .TITLE   SCR$VECTOR - Entry vectors for Screen Package
0000      2                    .IDENT   'V04-000'                        ; File: SCRVECTOR.MAR Edit: SBL1002
0000      3
0000      4          ;
0000      5          ;*********************************************************************
0000      6          ;*                                                                  *
0000      7          ;*   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                        *
0000      8          ;*   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.         *
0000      9          ;*   ALL RIGHTS RESERVED.                                           *
0000     10          ;*                                                                  *
0000     11          ;*   THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000     12          ;*   ONLY  IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE *
0000     13          ;*   INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER *
0000     14          ;*   COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000     15          ;*   OTHER PERSON. NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY *
0000     16          ;*   TRANSFERRED.                                                    *
0000     17          ;*                                                                  *
0000     18          ;*   THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE *
0000     19          ;*   AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT *
0000     20          ;*   CORPORATION.                                                    *
0000     21          ;*                                                                  *
0000     22          ;*   DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS *
0000     23          ;*   SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.         *
0000     24          ;*                                                                  *
0000     25          ;*                                                                  *
0000     26          ;*********************************************************************
0000     27          ;
0000     28
0000     29          ;++
0000     30          ; FACILITY: Terminal-independent Screen Procedures
0000     31          ;
0000     32          ; ABSTRACT:
0000     33          ;
0000     34          ;        This module contains the entry vector definitions for the
0000     35          ;        Run-Time Library Terminal-Independent Screen Handling Procedures
0000     36          ;
0000     37          ; ENVIRONMENT: Runs at any access mode, AST Reentrant
0000     38          ;
0000     39          ; AUTHOR: Steven B. Lionel, CREATION DATE: 26-Oct-1981
0000     40          ;
0000     41          ; MODIFIED BY:
0000     42          ;
0000     43          ; 1-001 - Original. SBL 26-Oct-1981
0000     44          ; 1-002 - Change PSECT name to $$VECTOR so that it sorts first alphabetically.
0000     45          ;         SBL 4-Dec-1981
0000     46          ;--
```

```
0000    48                  .SBTTL  DECLARATIONS
0000    49  ;
0000    50  ; LIBRARY MACRO CALLS:
0000    51  ;
0000    52  ;       NONE
0000    53  ;
0000    54  ; EXTERNAL DECLARATIONS:
0000    55  ;
0000    56          .DSABL  GBL             ; Force all external symbols to be declared
0000    57  ;
0000    58  ; MACROS:
0000    59  ;
0000    60
0000    61  ;+
0000    62  ; Macro to define an entry vector.  If NAME does not have a .ENTRY of
0000    63  ; its own, the corresponding .ENTRY name must be specified as ALTMASK.
0000    64  ;-
0000    65
0000    66          .MACRO  VECTOR  NAME, ALTMASK
0000    67          .EXTRN          NAME
0000    68          .TRANSFER       NAME
0000    69          .IF BLANK ALTMASK
0000    70          .MASK           NAME
0000    71          .IFF
0000    72          .MASK           ALTMASK
0000    73          .ENDC
0000    74          JMP             NAME+2
0000    75          .ENDM
0000    76
0000    77          .LIST   MEB                     ; generate listing for code generated
0000    78
0000    79  ;
0000    80  ; EQUATED SYMBOLS:
0000    81  ;
0000    82  ;       NONE
0000    83  ;
0000    84  ; OWN STORAGE:
0000    85  ;
0000    86  ;
0000    87  ;       NONE
0000    88  ;
0000    89  ; PSECT DECLARATIONS:
0000    90  ;
00000000 91          .PSECT $$VECTOR PIC, USR, CON, REL, LCL, SHR, -
0000    92                  EXE, RD, NOWRT, LONG
0000    93
```

```
                      0000        95  ;+
                      0000        96  ; Define vectored entry points for the screen package.
                      0000        97  ;-
                      0000        98
                      0000        99          VECTOR  LIB$ERASE_PAGE
           0000'      0000                    .MASK           LIB$ERASE_PAGE
00000002'EF  17       0002                    JMP             LIB$ERASE_PAGE+2
                      0008       100          VECTOR  SCR$ERASE_PAGE
           0000'      0008                    .MASK           SCR$ERASE_PAGE
00000002'EF  17       000A                    JMP             SCR$ERASE_PAGE+2
                      0010       101          VECTOR  SCR$ERASE, SCR$ERASE_PAGE        ; Obsolete
           0000'      0010                    .MASK           SCR$ERASE_PAGE
00000002'EF  17       0012                    JMP             SCR$ERASE+2
                      0018       102
                      0018       103          VECTOR  LIB$ERASE_LINE
           0000'      0018                    .MASK           LIB$ERASE_LINE
00000002'EF  17       001A                    JMP             LIB$ERASE_LINE+2
                      0020       104          VECTOR  SCR$ERASE_LINE
           0000'      0020                    .MASK           SCR$ERASE_LINE
00000002'EF  17       0022                    JMP             SCR$ERASE_LINE+2
                      0028       105
                      0028       106          VECTOR  LIB$PUT_LINE
           0000'      0028                    .MASK           LIB$PUT_LINE
00000002'EF  17       002A                    JMP             LIB$PUT_LINE+2
                      0030       107          VECTOR  SCR$PUT_LINE
           0000'      0030                    .MASK           SCR$PUT_LINE
00000002'EF  17       0032                    JMP             SCR$PUT_LINE+2
                      0038       108
                      0038       109          VECTOR  LIB$SET_CURSOR
           0000'      0038                    .MASK           LIB$SET_CURSOR
00000002'EF  17       003A                    JMP             LIB$SET_CURSOR+2
                      0040       110          VECTOR  SCR$SET_CURSOR
           0000'      0040                    .MASK           SCR$SET_CURSOR
00000002'EF  17       0042                    JMP             SCR$SET_CURSOR+2
                      0048       111
                      0048       112          VECTOR  LIB$PUT_SCREEN
           0000'      0048                    .MASK           LIB$PUT_SCREEN
00000002'EF  17       004A                    JMP             LIB$PUT_SCREEN+2
                      0050       113          VECTOR  SCR$PUT_SCREEN
           0000'      0050                    .MASK           SCR$PUT_SCREEN
00000002'EF  17       0052                    JMP             SCR$PUT_SCREEN+2
                      0058       114
                      0058       115          VECTOR  LIB$GET_SCREEN, SCR$GET_SCREEN
           0000'      0058                    .MASK           SCR$GET_SCREEN
00000002'EF  17       005A                    JMP             LIB$GET_SCREEN+2
                      0060       116          VECTOR  SCR$GET_SCREEN
           0000'      0060                    .MASK           SCR$GET_SCREEN
00000002'EF  17       0062                    JMP             SCR$GET_SCREEN+2
                      0068       117
                      0068       118          VECTOR  LIB$DOWN_SCROLL, SCR$DOWN_SCROLL
           0000'      0068                    .MASK           SCR$DOWN_SCROLL
00000002'EF  17       006A                    JMP             LIB$DOWN_SCROLL+2
                      0070       119          VECTOR  SCR$DOWN_SCROLL
           0000'      0070                    .MASK           SCR$DOWN_SCROLL
00000002'EF  17       0072                    JMP             SCR$DOWN_SCROLL+2
                      0078       120
                      0078       121          VECTOR  LIB$UP_SCROLL, SCR$UP_SCROLL
```

```
                 0000' 0078                          .MASK           SCR$UP_SCROLL
     00000002'EF    17  007A                          JMP            LIB$UP_SCROLL+2
                        0080   122             VECTOR  SCR$UP_SCROLL
                 0000' 0080                          .MASK           SCR$UP_SCROLL
     00000002'EF    17  0082                          JMP            SCR$UP_SCROLL+2
                        0088   123
                        0088   124             VECTOR  LIB$SET_SCROLL
                 0000' 0088                          .MASK           LIB$SET_SCROLL
     00000002'EF    17  008A                          JMP            LIB$SET_SCROLL+2
                        0090   125             VECTOR  SCR$SET_SCROLL
                 0000' 0090                          .MASK           SCR$SET_SCROLL
     00000002'EF    17  0092                          JMP            SCR$SET_SCROLL+2
                        0098   126
                        0098   127             VECTOR  LIB$SET_BUFFER, SCR$SET_BUFFER
                 0000' 0098                          .MASK           SCR$SET_BUFFER
     00000002'EF    17  009A                          JMP            LIB$SET_BUFFER+2
                        00A0   128             VECTOR  SCR$SET_BUFFER
                 0000' 00A0                          .MASK           SCR$SET_BUFFER
     00000002'EF    17  00A2                          JMP            SCR$SET_BUFFER+2
                        00A8   129
                        00A8   130             VECTOR  LIB$PUT_BUFFER
                 0000' 00A8                          .MASK           LIB$PUT_BUFFER
     00000002'EF    17  00AA                          JMP            LIB$PUT_BUFFER+2
                        00B0   131             VECTOR  SCR$PUT_BUFFER
                 0000' 00B0                          .MASK           SCR$PUT_BUFFER
     00000002'EF    17  00B2                          JMP            SCR$PUT_BUFFER+2
                        00B8   132
                        00B8   133             VECTOR  LIB$SCREEN_INFO
                 0000' 00B8                          .MASK           LIB$SCREEN_INFO
     00000002'EF    17  00BA                          JMP            LIB$SCREEN_INFO+2
                        00C0   134             VECTOR  SCR$SCREEN_INFO
                 0000' 00C0                          .MASK           SCR$SCREEN_INFO
     00000002'EF    17  00C2                          JMP            SCR$SCREEN_INFO+2
                        00C8   135
                        00C8   136             VECTOR  LIB$SET_OUTPUT
                 0000' 00C8                          .MASK           LIB$SET_OUTPUT
     00000002'EF    17  00CA                          JMP            LIB$SET_OUTPUT+2
                        00D0   137             VECTOR  SCR$SET_OUTPUT
                 0000' 00D0                          .MASK           SCR$SET_OUTPUT
     00000002'EF    17  00D2                          JMP            SCR$SET_OUTPUT+2
                        00D8   138
                        00D8   139             VECTOR  LIB$STOP_OUTPUT, SCR$STOP_OUTPUT
                 0000' 00D8                          .MASK           SCR$STOP_OUTPUT
     00000002'EF    17  00DA                          JMP            LIB$STOP_OUTPUT+2
                        00E0   140             VECTOR  SCR$STOP_OUTPUT
                 0000' 00E0                          .MASK           SCR$STOP_OUTPUT
     00000002'EF    17  00E2                          JMP            SCR$STOP_OUTPUT+2
                        00E8   141
                        00E8   142             .END                                       ; End of module SCR$VECTOR
```

F 13

SCR$VECTOR   - Entry vectors for Screen Package   16-SEP-1984 02:16:59 VAX/VMS Macro V04-00   Page 5
Symbol table   5-SEP-1984 04:43:38 [VMSLIB.SRC]SCRVECTOR.MAR;1   (3)

```
LIB$DOWN_SCROLL    ********  X  01
LIB$ERASE_LINE     ********  X  01
LIB$ERASE_PAGE     ********  X  01
LIB$GET_SCREEN     ********  X  01
LIB$PUT_BUFFER     ********  X  01
LIB$PUT_LINE       ********  X  01
LIB$PUT_SCREEN     ********  X  01
LIB$SCREEN_INFO    ********  X  01
LIB$SET_BUFFER     ********  X  01
LIB$SET_CURSOR     ********  X  01
LIB$SET_OUTPUT     ********  X  01
LIB$SET_SCROLL     ********  X  01
LIB$STOP_OUTPUT    ********  X  01
LIB$UP_SCROLL      ********  X  01
SCR$DOWN_SCROLL    ********  X  01
SCR$ERASE          ********  X  01
SCR$ERASE_LINE     ********  X  01
SCR$ERASE_PAGE     ********  X  01
SCR$GET_SCREEN     ********  X  01
SCR$PUT_BUFFER     ********  X  01
SCR$PUT_LINE       ********  X  01
SCR$PUT_SCREEN     ********  X  01
SCR$SCREEN_INFO    ********  X  01
SCR$SET_BUFFER     ********  X  01
SCR$SET_CURSOR     ********  X  01
SCR$SET_OUTPUT     ********  X  01
SCR$SET_SCROLL     ********  X  01
SCR$STOP_OUTPUT    ********  X  01
SCR$UP_SCROLL      ********  X  01
```

```
+-----------------+
! Psect synopsis !
+-----------------+
```

```
PSECT name                    Allocation          PSECT No.  Attributes
----------                    ----------          ---------  ----------

.   ABS   .                   00000000 (    0.)   00 (   0.)  NOPIC  USR   CON   ABS   LCL  NOSHR  NOEXE  NORD  NOWRT  NOVEC  BYTE
$$VECTOR                      000000E8 (  232.)   01 (   1.)    PIC  USR   CON   REL   LCL   SHR    EXE    RD   NOWRT  NOVEC  LONG
```

```
+----------------------+
! Performance indicators !
+----------------------+
```

```
Phase                    Page faults   CPU Time      Elapsed Time
-----                    -----------   --------      ------------
Initialization                    32   00:00:00.10   00:00:00.64
Command processing               118   00:00:00.57   00:00:03.06
Pass 1                            95   00:00:00.93   00:00:02.81
Symbol table sort                  0   00:00:00.01   00:00:00.01
Pass 2                            50   00:00:00.46   00:00:00.88
Symbol table output                4   00:00:00.03   00:00:00.24
Psect synopsis output              2   00:00:00.02   00:00:00.02
Cross-reference output             0   00:00:00.00   00:00:00.00
Assembler run totals             303   00:00:02.12   00:00:07.66
```

The working set limit was 900 pages.
8228 bytes (17 pages) of virtual memory were used to buffer the intermediate code.

There were 10 pages of symbol table space allocated to hold 29 non-local and 0 local symbols.
142 source lines were read in Pass 1, producing 12 object records in Pass 2.
1 page of virtual memory was used to define 1 macro.

```
                                      +------------------------------+
                                      ! Macro library statistics !
                                      +------------------------------+
```

Macro library name   Macros defined
-----------------   --------------
_255DUA28:[SYS.OBJ]LIB.MLB;1   0
_255DUA28:[SYSLIB]STARLET.MLB;2   0
TOTALS (all libraries)   0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/DISA=TRACE/LIS=LIS$:SCRVECTOR/OBJ=OBJ$:SCRVECTOR MSRC$:SCRVECTOR/UPDATE=(ENH$:SCRVECTOR)+EXECML$/LIB

SYSGMSG
LIS

SCRINPUT
LIS

SCRMISC
LIS

SCRREQ
LIS

SETPRIV
LIS

SHRMSG
LIS

TPARSE
LIS

SCRCTB
LIS

SCRVECTOR
LIS

SSMSG
LIS